Oracle Darwin

Darwin Installation and Administration

Release 3.7 for Solaris

November 2000

Part No. A86034-01

ORACLE®

Darwin Installation and Administration, Release 3.7 for Solaris

Part No. A86034-01

Contents

5 Oracle Connectivity

6 Installing the Darwin Client

7 Starting and Stopping Darwin

8 Administering Darwin Projects

A Installing the SAS Conversion Utilities

B Server Configuration Files

C Upgrading Darwin

Index

Send Us Your Comments

Darwin Installation and Administration, Release 3.7 for Solaris

Part No. A86034-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the chapter, section, and page number (if available). You can send comments to us in the following ways:

- DARWINDOC@us.oracle.com
- FAX: 781-684-7738. Attn: Oracle Darwin
- Postal service:
  Oracle Corporation
  Oracle Darwin Documentation
  200 Fifth Avenue
  Waltham, Massachusetts 02451
  U.S.A.

If you would like a reply, please give your name, address, and telephone number below.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Darwin is a data mining application designed specifically to handle multiple gigabytes of data, and to provide answers to complex problems of data classification, prediction, and forecasting.

This manual describes how to install and administer Darwin Release 3.7 server software on selected Sun platforms running Solaris 2.7 and client software on PCs running Windows NT 4.00, Windows 95, or Windows 98. There is a separate manual that describes installing and administering Darwin Release 3.7 software on HP-UX.

Intended Audience

This manual is intended for experienced Solaris system administrators.

Structure

This manual contains eight chapters:

Chapter 1	**Overview** Provides an overview of the installation procedure.
Chapter 2	**Requirements** Lists the hardware and software requirements.
Chapter 3	**Installing the Darwin Server** Describes how to install the Darwin server software.
Chapter 4	**Configuring and Starting a Darwin Server** Describes how to configure and start Darwin servers.

Chapter 5	**Database Connectivity** (optional) Describes how to set up the environment for database connectivity.
Chapter 6	**Installing the Darwin Client** Describes how to install Darwin client software.
Chapter 7	**Starting and Stopping Darwin** Describes how to start and stop Darwin servers.
Chapter 8	**Administering Darwin Projects** Describes some administrative functions associated with Darwin projects, and describes how to create a project.

In addition, there are three appendixes that cover supplementary topics:

Appendix A	**Installing the SAS Conversion Utilities** (optional) Explains how to install `sas2darwin` and `darwin2sas`, utilities used for converting between Darwin datasets and SAS files.
Appendix B	**Server Configuration Files** Explains the meaning of entries in a Darwin configuration file.
Apprendix C	**Upgrading Darwin** (optional) Summarizes the changes required for moving to Darwin 3.7 from an earler release.

Related Documents

Darwin documentation is distributed on the documentation CD in PDF and HTML formats.

The complete Darwin documentation set includes

- *Darwin New Features, Release 3.7*. Describes the features introduced at Release 3.5, Release 3.6, and Release 3.7.

  This manual is a revision of *Darwin 3.6 New Features*, which described the functionality introduced at Release 3.5 and Release 3.6. *Darwin New Features, Release 3.7* contains updated information about the functionality introduced at

Releases 3.5 and 3.6 plus new material describing functionality introduced at Release 3.7.

If you are upgrading from 3.6.1, you can discard the manual *Darwin 3.6 New Features*; *Darwin New Features, Release 3.7* supersedes it.

- *Darwin 3.7 Release Notes for Solaris.* Describes the release, documents any problems or bugs in the software, and describes any changes that occurred in the software after the manuals were finalized. There are separate release notes for Solaris and for HP-UX.
- For Solaris system administrators: *Darwin Installation and Administration, Release 3.7 for Solaris* (this manual). Describes how to install server and client software. Changes that developed after this manual was finalized are described in the release notes. There are separate installation / administration guides for Solaris and for HP-UX.
- *Using Darwin, Release 3.0.1.* A how-to manual; describes the user interface and provides detailed instructions for using it. (*Using Darwin* describes all the features available at Release 3.0.1; together with *Darwin New Features, Release 3.7* and the release notes, you have a complete description of the user interface at Release 3.7.)
- *Darwin Reference, Release 3.0.1* (companion volume to *Using Darwin*). Introduces data mining and Darwin; provides background and conceptual material on datasets, Darwin tools, and analyses.

Documentation CD

All the Darwin documentation is available in HTML and PDF format on the documentation CD; you can read or print documentation directly from the CD.

To view the PDF files, you will need

- Adobe Acrobat Reader 3.0 or later, which you can download from `www.adobe.com`.

To view the HTML files, you will need

- Netscape 3.x or later, or
- Internet Explorer 4.0 or later

Darwin Online Help

Darwin includes extensive online help that can be summoned from a list of contents, from the **Help** tab, and from **Help** buttons or the **F1** key on dialog windows. For correct display of Darwin's online help, you need Internet Explorer 4.x. If you do not have it, you can download it from `www.microsoft.com`.

Conventions

The following conventions are used in this manual:

Convention	Meaning
boldface	Darwin commands, menu names, menu items, names of dialogs and screens.
Project > New File	Indicates the path for a command. The example shown means on the **Project** menu, click the **New File** command.
`code`	Data fields and values, special characters, etc., examples of files, data, filenames, and pathnames.
`italics`	Argument names and placeholders in command formats.
`% user input` `system output`	In interactive examples, user input is shown in bold typewriter, and system output is shown in regular typewriter.

1

Overview

This manual describes installation and administrative procedures for Darwin Release 3.7 on selected Sun platforms running Solaris 2.6 or 2.7 with client software running on Microsoft Windows NT 4.00, Windows 95, or Windows 98.

1.1 Installation Overview

Darwin is distributed on three CDs:

- the Darwin server CD, formatted to be read by Solaris
- the Darwin client CD, formatted to be read by Windows NT/95/98
- the Darwin documentation CD, formatted to be read by Windows NT/95/98

Each CD contains a file `README.txt` that describes the contents of the CD and contains other useful information.

Note: You *must* install the same version of the Darwin client and server.

To install Darwin, follow these steps:

- Confirm that all requirements are met (Chapter 2)
- Install the Darwin server on Solaris (Chapter 3)
- Configure and start at least one Darwin server (Chapters 4 and 8)

> **Note:** Darwin server configurations created with earlier releases of Darwin will no longer work properly; see Appendix C.

- Set up for database connectivity (optional) (Chapter 5)

- Install Darwin clients on Windows NT/95/98 (Chapter 6)
- Start and stop Darwin servers (Chapter 7)
- Create a Darwin project (Chapter 8)

The chapters in this manual cover these steps in order, beginning with Chapter 2.

1.2 Where to Go for More Information

See the preface for a complete list of related Darwin documentation.

Darwin also provides extensive online help that can be summoned from pulldown menu and a **Help** tab and from **Help** buttons or the **F1** key on dialog windows.

2

Requirements

This chapter describes the hardware, software, disk space, network, and other requirements for the Darwin Release 3.7 for Solaris client and server.

2.1 Hardware and Operating System Requirements

Darwin 3.7 server software runs on UNIX (Solaris 2.7 or HP-UX 11.0).

The Darwin 3.7 server software for Solaris runs on the following hardware platforms:

- Sun Microsystems UltraSPARC workstations running Solaris 2.7
- Sun Microsystems Enterprise Servers, including the Enterprise 10000, running Solaris 2.7

> **Note:** Darwin on an Enterprise 10000 requires a static IP address; see Section 2.1.1.

If you will run Darwin on a multi-node Solaris server, see Section 2.7 for information about hardware requirements.

The Darwin 3.7 client (user interface) software runs on the following platforms:

- personal computers running Microsoft Windows NT 4.00, Microsoft Windows 95, or Microsoft Windows 98

Darwin has not been tested with Windows 2000.

On any of the PC platforms, Microsoft Internet Explorer 4.x is required to display online help. Without Internet Explorer, Darwin functions properly but online help is not available.

We recommend that the PCs that Darwin clients run on should have a CPU that is at least Pentium 100 MHz or equivalent and should have at least 16 Mbytes of RAM for Windows 95 and Windows 98 systems and 32 Mbytes of RAM for Windows NT systems.

2.1.1 Sun Enterprise 10000 Requirement

On an Enterprise 10000, Darwin requires a static IP address or hostname. Darwin will not run on a partition where the IP address is assigned dynamically, as, for example, in some partitions that support failover.

2.1.2 Solaris Requirements and Recommendations

2.1.2.1 Solaris Requirement for Multi-Node Systems (Optional)

If you plan to run Darwin on a multi-node system (Ultra Enterprise server or SMP), you must edit the `/etc/system` file on the system before you install Darwin server. These changes are required to enable shared libraries. If you are not going to install Darwin on a multi-node system, you do not have to make these changes.

To make these changes, log in to the multi-node Solaris system as root. Add the following entries to the end of the file `/etc/system`:

```
set pt_cnt=1024
set npty=1024
set sadcnt=2048
set nautopush=1024
set dosynctodr=0
set shmsys:shminfo_shmmax=2120000000
set shmsys:shminfo_shmmni=200
set shmsys:shminfo_shmseg=200
```

After you've made the changes, reboot your system before installing Darwin server software. The changes to `/etc/system` don't take effect until you reboot.

2.1.2.2 Oracle OCI Library

Darwin 3.7 uses Oracle Call Interface (OCI) to connect to Oracle databases. The appropriate OCI library is bundled with Darwin. This library is required even if no Darwin servers that support database connectivity exist. Every Darwin server contains a definition of the default value for the environment variable `ORACLE_HOME`.

2.2 Disk Space Requirements

2.2.1 Server Disk Space Requirements

The Darwin server requires approximately 30 Mbytes of disk space. This is the amount of space the Darwin software itself requires, and does not include space required for any database files or swap space. If you install Darwin server as described in Chapter 3, you will also need 600 Mbytes of free space in `/tmp`.

2.2.2 Client Disk Space Requirements

The Darwin client requires approximately 16 Mbytes of disk space on Windows NT, Windows 95, or Windows 98.

2.3 Client Requirements

A PC running Darwin client requires the following:

- A network connection to the server (Section 2.3.1)
- `English (United States)` regional settings (Section 2.3.2)
- Microsoft Internet Explorer 4.x for online help (Section 2.3.3)
- Microsoft Excel for graphing (Section 2.3.4)

2.3.1 Network Connection Requirement

There must be direct network connection (via TCP/IP) from any personal computer running Darwin client software to Solaris.

2.3.2 Regional Settings

The regional settings on any PC running the Darwin client must be `English (United States)`. If you specify any other language, the Darwin client will not operate properly.

2.3.3 Microsoft Internet Explorer Requirement

Microsoft Internet Explorer 4.x is required for the operation of Darwin online help. Internet Explorer 4.x must be installed on the PC *before* you install the Darwin client.

Note: If Internet Explorer is not installed on the PC where Darwin client is installed, no online help will be available. (**Help** buttons will not work; the **F1** key will not display help; the **Help** tab links will not work; the **Help** button on the tool bar will not work; all items in the **Help** menu, except for **Help > About Darwin** will not work.)

Except for the lack of online help, Darwin installed on a PC without Internet Explorer will work properly.

You can download Internet Explorer 4.x from the following site on the World Wide Web:

`http://www.microsoft.com/windows/ie/download/default.asp`

Click **Internet Explorer 4.01 Service Pack 2** under the heading **Other Downloads**. This takes you to the Internet Explorer 4.01 Service Pack 2 Download Page. Select the operating system on your PC, and then click the **Download Now** button.

Internet Explorer must be installed on your system; it does not have to be your default browser.

Darwin has not been tested with Internet Explorer 5.0, 5.01, or 5.5.

2.3.4 Microsoft Excel Requirement

Microsoft Excel is used by the Darwin client to create plots and graphs; one of the following is required:

- Microsoft Excel 95 for Windows 95, Version 7.0
- Microsoft Excel 97

Note: You must install Excel on the C Drive of the PC. Darwin cannot find Excel if it is installed on any other drive.

If Excel 95 or Excel 97 is not installed on your PC, you will not be able to create any plots or graphs using Darwin.

Darwin has not been tested with Office 2000.

2.4 Requirements for Code Generation

You can export Darwin models as C, C++, or Java code using the **Code Generation** command of the **Options** menu. You can generate model code for tree or net models, but not for match models.

Code generation is enabled by default; in previous releases of Darwin, a license was required. If you already have a license for code generation, you do not have to remove it; it is ignored.

2.5 SAS Conversion Utilities Hardware and Software Requirements

Note: If you do not plan to convert SAS files to/from Darwin datasets, you may skip this section.

The SAS conversion utilities run on UNIX (Solaris and HP-UX). They require the following software to be installed on the UNIX system on which you plan to do conversions:

- Conceptual Software, Inc., DBMS/COPY, version 7 or higher
- Perl, version 5.001 or higher

For information about installing the conversion utilities, see Appendix A of this manual; for information about using the utilities, see *Using Darwin*.

2.5.1 DBMS/COPY

You must obtain DBMS/COPY from

Conceptual Software, Inc.
9660 Hillcroft #510
Houston, TX 77096
USA
Telephone: 1-800-328-2686 or 1-713-721-4200

For information about ordering DBMS/COPY, see Conceptual Software's page on the World Wide Web at `www.conceptual.com`. The price (when this manual was published) is approximately $500 (US) for an individual license.

2.5.2 Perl

If Perl is not already installed on your Solaris system, you can obtain it free of charge from several locations. For details, see `www.perl.org` or `www.perl.com` on the World Wide Web. If you do not wish to compile and build Perl for your system, you can download a Solaris version of Perl directly from `www.sunfreeware.com`. The web page includes directions for downloading.

2.5.3 Disk Space for the Conversion Utilities

DBMS/COPY and the Perl scripts require approximately 6.7 Mbytes of disk space in addition to the space required by Darwin.

When the SAS conversion utilities convert a file, they create temporary files; these temporary files require at least as much space as the file being converted.

2.6 Database Connectivity Requirements

Note: If you are not planning to connect to Oracle databases, you may skip this section.

Darwin 3.7 supports connectivity to Oracle 8.1.6 databases using Oracle Call Interface (OCI). The required OCI library is bundled with Darwin. This library is required even if no Darwin servers support database connectivity.

Any Oracle database that Darwin connects to must satisfy the following requirements:

- A listener must be set up.
- Oracle must support single byte 8-bit ASCII.

Listener must be set up for Oracle to communicate with Darwin 3.7, even if Oracle is installed on the same system as Darwin server.

For information about setting up for database connectivity, see Chapter 5 of this manual; for information about using database connectivity, see *Using Darwin* and *Darwin New Features.*

2.6.1 Oracle Database Software Requirement

You will need following Oracle database product installed at your site:

- Oracle 8.1.6

For information about obtaining and installing these products, contact Oracle:

- Telephone: 1-800-ORACLE1
- Oracle's web site: `www.oracle.com/database/oracle8/`

2.6.2 Requirements for National Language Characters in Tables

Darwin supports connection to Oracle databases that support the WE8ISO8859P1 character set. You can import tables from and export datasets to such a database. Darwin does not support any character sets that are not a subset of this character set.

All Darwin servers are automatically configured to support the WE8ISO8859P1 character set.

If you connect to a database that supports some other character set, certain characters will not be available.

2.7 Darwin on Multi-Node Servers

This section describes things to keep in mind when running Darwin on multi-node servers.

If you do not plan to run Darwin on multi-node servers, you can ignore this section.

For information about running Darwin on Enterprise 10000 systems, see Section 2.1.1.

For information about settings that must be added to `/etc/system` before installing Darwin on a multi-node system, see Section 2.1.2.1.

When you run Darwin on a multi-node server, you may want to distribute any datasets that you use for model building. See Section 2.8 for information about distributed datasets.

2.7.1 Performance on Multi-Node Servers

Optimal performance of Darwin running on a multi-node server (SMP) is achieved when Darwin is the only program running on that SMP. If other programs are running on the SMP at the same time that Darwin is running, Darwin performance is poor.

Also, performance on a multi-node server is best when only one user at a time runs Darwin on any given set of CPUs. This restriction does not apply to multiple Darwin users on a single CPU system.

2.7.2 Model Building on Multi-Node Servers

When you build a model on a multi-node server, model building is always distributed, even if the datasets are serial. A serial dataset will be distributed as the operation begins, and re-serialized at its conclusion. The most efficient way to run a sequence of operations is to explicitly distribute the data, as described in Section 2.8, before launching the first operation.

2.7.3 Server Capacity Planning Guidelines

Below are some guidelines to be taken as "rules of thumb" in planning server capacity.

Sizing the Number of Server CPUs

Model building requires the most CPU resources to run the hundreds and thousands of calculations the algorithms perform against the training datasets. The testing and evaluating phases take less CPU resources, and scoring the full database with the deployed model takes the least CPU resources. Below is a Darwin performance table that you can use to derive the number of CPUs necessary to achieve a desired performance level; note that the numbers shown are valid for a particular dataset and not necessarily accurate for different datasets.

Model type	Number of CPUs			
	4	6	8	16
C&RT	23 min	15 min	12 min	6 min
Neural Net	93 min	55 min	47 min	24 min

Sizing Server RAM Memory

Model development is most efficient when the full training, testing, and evaluation datasets are in RAM memory at least individually.

To calculate Minimum memory size:

max number of records for modeling **x** max record size = ____MG / GB

Sizing Server Disk Storage Capacities

Allow 100 percent additional disk storage space over the raw database total storage requirements for database administration, transient model storage, and scored dataset storage.

2.8 Distributed Datasets

When you run Darwin on a multi-node server, you may want to create distributed datasets. Operations such as frequency counts or model building, performed on a distributed version of a dataset, are often faster than the same operations performed on the undistributed (serial) dataset.

All Darwin datasets are created, by default, as serial datasets, even on multi-node Darwin servers. If you wish to create a distributed version of a dataset, you must explicitly create it. If you create a dataset using either **Dataset > Create** or the **Text Import** Wizard, the resulting dataset is always serial.

If you want to perform an operation on a distributed dataset, you must distribute the dataset before you perform the operation. Once you distribute the dataset, you can save the distributed version for future use.

2.8.1 Distributing Datasets

Here are two ways to distribute a dataset; in both ways, Darwin creates a new dataset that is the distributed version of the original dataset:

- Use the **Randomize** transform to randomize the dataset. If you are connected to a distributed server, the transformed dataset is distributed in memory; you can save the transformed distributed dataset if you wish. The default name of the randomized distributed dataset is *`dataset-name`*`[randomize]`. (The original dataset that you applied the transformation to is still serial.)
- Select the dataset. Go to the **Datasets** tab of **Options > Advanced**; click the **Create Distributed** button. Darwin creates a distributed version of the dataset named *`dataset-name`*`[distributed]`; the dataset is automatically saved. The underlying file for the new dataset has the extension `.dst`.

For large datasets, either of these operations may take several minutes or more.

See also Section 2.8.2, below, if you need to move a distributed dataset.

For more information about either of these operations, see *Using Darwin*.

2.8.2 Moving Distributed Datasets

If you move your dataset between servers with different numbers of nodes, serialize the dataset first. Here's how:

- Click the dataset's name to select it.
- Click **Options > Advanced**.
- Click the **Datasets** tab.
- In the **Serial or Distributed** section, click **Create Serial**.

Darwin creates a serial version of the dataset named *`dataset-name`*`[serial]` (e.g., the serial version of `Ceil` is `Ceil[serial]`).

3

Installing Darwin Server

The first step in the Darwin installation process is installing the Darwin server software on Solaris, as described in this chapter. This chapter also includes an overview of upgrading from pervious versions of Darwin.

3.1 Upgrading Darwin

If Darwin is already on your system, you must stop (kill) any servers associated with Darwin (via `darwinconfig`) and remove the existing version of Darwin (via `pkgrm`) before you install the new version. If you wish, you can copy the current version of Darwin from `/opt/TMCdarwin` before removing the old version and installing the new one.

Before you deinstall pre-3.0 versions of Darwin, you should convert datasets and models, as described in Section C.1. Models and datasets created using Darwin 3.5 or later do not need any conversion.

Existing server configurations will not work properly with Darwin 3.7. For details, see Section C.3.

For more information on upgrading, see Appendix C.

3.2 Installing Darwin Server Software on Solaris

The Darwin server software is distributed as a tarfile on a CD. To install the Darwin server, follow these steps:

1. Install Oracle database software if you plan to configure servers that support database connectivity. You must set up a listener even if Oracle and Darwin are installed on the same machine. For more information, see Chapter 5.

2. Log in as root to the Solaris system where the Darwin server software will reside.

3. If any Darwin servers are already installed on the system and are running, stop each one using the following command, as described in Section 4.4.7

   ```
   /opt/TMCdarwin/etc/darwinconfig stop server-name
   ```

   where *server-name* is the name of the server.

4. If Darwin server software is installed on the system, remove it using the command

   ```
   pkgrm TMCdarwin
   ```

5. Make sure that there is enough space for the server software; you will need

   - 30 Mbytes in `/opt`
   - 60 Mbytes in `/tmp`

6. If the server software will not fit into `/tmp`, use some other directory; substitute the name of the other directory for `/tmp` in the steps that follow.

7. Make sure that there is no copy of the Darwin server software in the directory `/tmp`.

8. Insert the Darwin Server CD in a drive attached to the Solaris system where the servers will reside; copy the files from the CD to `/tmp` and extract them from the tarfile:

   ```
   cd /tmp
   tar xvf /cdrom/cdrom0/Darwin37Sol.tar
   ```

   where *cdrom* is the name of the CD drive.

9. To install Darwin, use the command `pkgadd`:

   ```
   pkgadd -d /tmp TMCdarwin
   ```

   Answer any questions that the command asks. This command creates the directory `/opt/TMCdarwin` (if it doesn't already exist).

Note: If `/opt` is not a distinct partition and is not large enough to contain Darwin, the Darwin software will be dumped into the root partition and may fill the root partition. A simple solution is, before running `pkgadd`, to create a symbolic link to `/opt/TMCdarwin` from a directory in a partition that has enough space for the Darwin software.

10. If you plan to connect to Oracle databases, edit the file `/opt/TMCdarwin/network/admin/tnsnames.ora`, as described in Chapter 5.
11. Check the installation, as described in Section 3.3.
12. Configure at least one Darwin server, as described in Chapter 4. Start any servers that you configure. Do not restart any server configurations created using earlier version of Darwin until you have corrected them.

3.3 Checking the Server Installation

Follow these steps to start the Darwin server `demo-server` to verify that the Darwin server installation was successful:

1. Log in as root.
2. Start the **`demo-setup`** script:

   ```
   cd /opt/TMCdarwin/demo
   sh demo-setup
   ```

3. Select **`1`** to start a server on Solaris. If you start the server on a port other than the default port 1022, note the port number. (When you configure Darwin servers, you should not use this port number.)
4. If the command is successful, you get the message

   ```
   Server started
   ```

   You can verify that the server is running using the UNIX `ps` command:

   ```
   ps -ef|grep darwin
   ```

4

Configuring Darwin Servers

After you have installed the Darwin server software, you must configure and start one or more Darwin servers, as described in this chapter. If you plan to connect to a database, see Chapter 5.

> **Note:** Darwin server configurations created using releases of Darwin earlier than Darwin 3.7 will not work with Darwin 3.7. See Section 4.3.2 for a summary of changes.

Basically, configuring a Darwin server means using the `darwinconfig` command and its subcommands to add, remove, describe, start, stop, etc., Darwin servers. In addition, the `darwinconfig` command creates and maintains several configuration files for both client and server.

Section 4.1 provides a brief overview of Darwin's client-server design, and shows how the various Darwin components interact with each other.

This rest of this chapter explains

- Darwin configuration files (Section 4.2)
- what you need to do before you configure a server (Section 4.3)
- how to add, remove, and configure a server, using `darwinconfig` and its subcommands (Section 4.4)

For examples of configuration files and for an explanation of the entries in a server configuration file, see Appendix B.

4.1 Darwin Servers

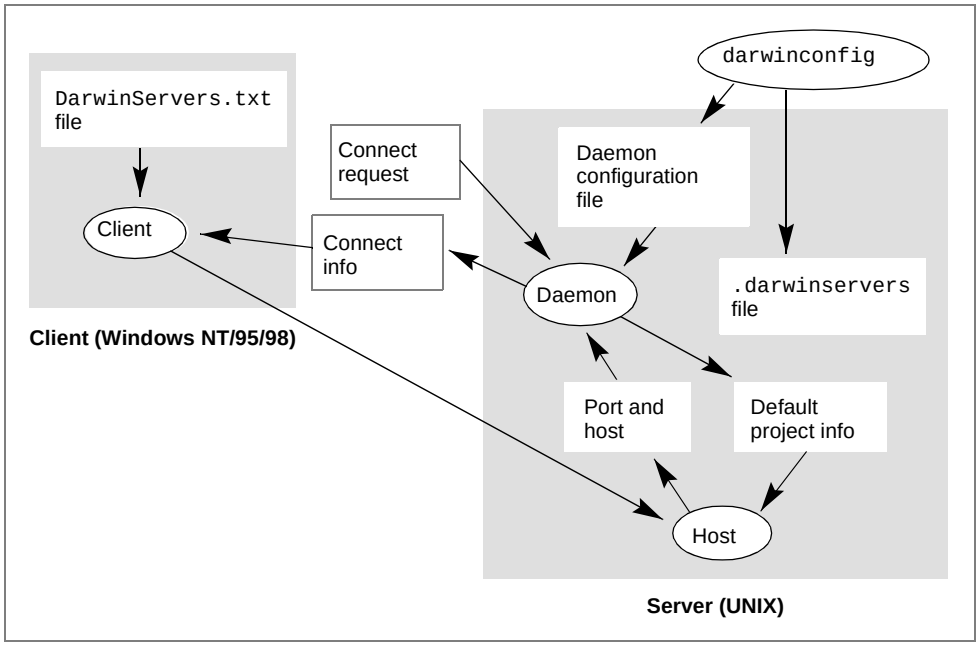
Darwin uses a client-server design, a design that separates client and server functions. Typically, a client workstation provides the user interface, does some of the processing, and communicates with a server. The server does most of the processing and handles requests from the client.

Darwin's client-server design distributes the computing as follows: the Darwin client presents and updates the user interface and the graphics, and also handles some local computation. The Darwin server runs the Darwin data mining algorithms.

The Darwin server is composed of the necessary Darwin executable files, a daemon, and a configuration file created using the `darwinconfig` command. Each daemon runs a particular Darwin executable in a particular way.

Figure 4–1 shows how Darwin components interact.

Figure 4–1 Interaction of Darwin components.



4.2 Darwin Configuration Files

The server's configuration file contains all the information necessary to run the server and set up projects. Appendix B contains sample configuration files for the Solaris platforms that Darwin supports at Release 3.7.

Note: All file locations assume a standard installation in which the software is installed in `/opt/TMCdarwin`, the default location. If the software is installed in some other directory, correct the pathname accordingly.

The command `darwinconfig`, described in Section 4.4, creates and maintains the following files:

- a configuration file named `.darwinservers`, which covers all configured Darwin servers at your site. The `.darwinservers` file is created in `/opt/TMCdarwin/etc`. Section 4.2.1 describes the `.darwinservers` file.
- a configuration file for each server on the system. Each server's configuration file has the same name as its server. These configuration files are saved on the UNIX server in `/opt/TMCdarwin/etc`. Appendix B shows sample configuration files for single and multi-node servers with and without database connectivity.

4.2.1 .darwinservers File

The `.darwinservers` file lists all configured Darwin servers. In a standard configuration created by the `darwinconfig` command, the `.darwinservers` file is automatically created and updated in `/opt/TMCdarwin/etc` on UNIX.

Each line in the `.darwinservers` file describes a single server. Each line has five parts:

- the keyword `server`
- the Darwin server's name (the name specified in the `name` line of the server's configuration file)
- the hostname of the system on which the server runs (the name specified in the `server` line of the server's configuration file)
- the port to be used by the server (matching the `port` entry in the configuration file for the server)
- a description of the Darwin server, within quotation marks

Notice that the server's name, the name of the system on which the server runs, and the port number must match the values in the server's configuration file (see examples in Appendix B). Here is a sample `.darwinservers` file for the sample server configuration files for servers `eval-1` and `Ring`:

```
server eval-1 test.mysite.com 1066 "Test System 1"
server Ring mysmpserver 1111 "SMP test system"
```

4.2.2 DarwinServers.txt File

A related file, `DarwinServers.txt`, which you create (see Section 6.2.1), resides on each user's C drive on Windows NT or Windows 95/98:

```
C:\Program Files\Tmc\Darwin Windows Client\DarwinServers.txt
```

This file contains a list of servers to which the client can connect. You may wish to copy `.darwinservers` to this file. Darwin does not update `DarwinServers.txt`; you must update it by hand.

When you install a Darwin client, you specify the file `DarwinServers.txt` as the source of the list of servers to which users can connect.

When a user opens a Darwin session, the Darwin client displays, on the login screen, a list of server names and descriptions, taken from the `DarwinServers.txt` file. (Using the sample file shown above, the login screen would offer the user the choice of servers eval-1 and Ring.) The user selects a server from this list; Darwin then attempts to connect the user to that server.

4.3 Before Using darwinconfig

Before you can use the `darwinconfig` command, you must

- be logged in as root to the UNIX system on which the server will run
- determine the UNIX directory where Darwin server software is installed (see Section 4.3.1, below)

4.3.1 Changing Defaults

If you have a nonstandard installation (that is, Darwin is installed in some directory other than `/opt/TMCdarwin`), you must modify the default environment variables that `darwinconfig` adds to all configurations. See Section 4.4.11.1 for details.

4.3.2 Upgrading Existing Configuration Files

For more information about converting existing server configurations, see Appendix C.

Darwin 3.7 supports architectures and executables different from those supported by earlier versions of Darwin. Server configurations created using earlier versions of Darwin will not work properly with Darwin 3.7.

The supported architectures, **`sparc`** and **`hpux`**, do not have the same meaning that they had in versions of Darwin prior to 3.6.1.

In addition, the name of the Darwin server executable changed. Configurations created prior to Darwin 3.6.1 that specify any other executable cannot be started; users cannot connect to them.

Configurations created with Darwin 3.6.1 or earlier do not have the correct environment variables.

Configurations supporting database connectivity created for any version of Darwin prior to Darwin 3.7 will not work. See Chapter 5 for details.

You must create a new configuration for each Darwin server.

Note that you *cannot* modify existing configurations files; instead you must create new configurations. See Section 4.4.13 for details.

4.4 Configuring and Managing Darwin Servers

This section describes the `darwinconfig` subcommands and related topics. The subcommands are described in Sections 4.4.1 through 4.4.10. Also discussed is

- Modifying and adding environment variables Section 4.4.11)
- Modfying shared configurations (Section 4.4.12)
- Modifying configuration files (Section 4.4.13)

The command `darwinconfig` runs on UNIX (Solaris or HP-UX); you *must* be logged in as root to perform tasks such as creating configuration files or starting and stopping servers.

Note: If you are not logged in as root when you create configuration files and start or stop servers, the operations will not be performed correctly.

To use the `darwinconfig` command, cd (on UNIX) to the appropriate directory and issue the `darwinconfig` command:

```
# cd /opt/TMCdarwin/etc
# ./darwinconfig
Darwin SCU >
```

The `Darwin SCU >` prompt indicates that you are running `darwinconfig`. The command is interactive; you are prompted for information required for the particular subcommand you are using.

The `darwinconfig` command provides these subcommands:

- **`list`** lists the available Darwin servers (Section 4.4.1).
- **`add`** adds a new Darwin server (Section 4.4.2).
- **`remove`** removes an existing Darwin server (Section 4.4.3).
- **`describe`** displays a description of a Darwin server (Section 4.4.4).
- **`verify`** verifies the configuration of a Darwin server (Section 4.4.5).
- **`start`** starts an existing Darwin server (Section 4.4.6).
- **`stop`** stops a running Darwin server (Section 4.4.7).
- **`save`** saves a new configuration file (Section 4.4.8).
- **`quit`** quits `darwinconfig` without saving changes (Section 4.4.9).
- **`exit`** exits `darwinconfig` after saving changes (Section 4.4.9).
- **`help`** or**`?`** prints a list of the subcommands and their definitions (Section 4.4.10).

4.4.1 Listing Available Servers

Use `list` to display a list of all of the available servers (i.e., all servers in the `.darwinservers` file described in Section 4.2.1). For each server, `list` displays the name, the architecture, the hostname, whether auto restart is specified, and whether the server is running.

An example of output to `list`:

```
Darwin SCU > list
Available servers:
Name         Arch    Hostname              Auto Restart Running
DarthSerial  sparc   darth-smp.think.com   Yes          Yes
DarthSmp-1   sparc   darth-smp.think.com   No           Yes
DarthSmp-2   sparc   darth-smp.think.com   Yes          Yes
DarthSmp-4   sparc   darth-smp.think.com   Yes          Yes
DarthDB      sparc   darth-smp.think.com   Yes          No
```

4.4.2 Adding and Configuring a Server

Use `add` to create and configure a new Darwin server. The subcommand `add` prompts you for information about server characteristics and uses your answers to create a configuration file for the server.

NOTE: The port number that you specify during server configuration must be greater than 1024.

After you've answered the required questions, `darwinconfig` creates a configuration file, and displays it for your review. Check it carefully, especially the environment variables.

`darwinconfig` automatically adds default environment variables to the server configuration. If Darwin server software is installed in the default location `/opt/TMCdarwin`, the default values will be appropriate for all servers. If Darwin server software is not installed in the default location, you will have to change the default definitions by replacing `/opt/TMCdarwin` with the pathname where Darwin server software is installed. Section 4.4.11.1 describes how to modify the environment variables.

If `add` finds problems with your answers, it generates messages about the problems. Remember that no two servers can have the same name.

Note: The server's configuration file is not automatically saved; to save it, use `save` (Section 4.4.8) or `exit` (Section 4.4.9). You must save a server's configuration file before you can start that server.

When you create and configure your first Darwin server, a `.darwinservers` file is created. Thereafter, the `.darwinservers` file is updated whenever a new server is created or a server is removed. See Section 4.2.1 for more information about the `.darwinservers` file.

On Windows NT or Windows 95/98, the file `DarwinServers.txt` in `C:\Program Files\Tmc\Darwin Windows Client` contains a list of servers to which the client can connect. When you are finished configuring servers, you can use the information in `.darwinservers` to create this file. Since Darwin does not update `DarwinServers.txt`, the user or system administrator must update it.

For examples of complete configuration files, see Appendix B; Appendix B also contains detailed descriptions of the meaning of the entries in the configuration file. You will find it useful to have Appendix B available as you answer the questions generated by `add`.

4.4.3 Removing a Server

To remove a Darwin server, first stop the server using the `stop` subcommand, described in Section 4.4.7. Use `remove` to remove an existing configuration file from `/opt/TMCdarwin/etc`. The server is then no longer available to Darwin clients, and the `.darwinservers` file is automatically updated.

The `DarwinServers.txt` file is not automatically updated; you must update that file yourself from the Windows NT or Windows 95/98 file:

```
C:\Program Files\Tmc\Darwin Windows Client\DarwinServers.txt
```

4.4.4 Describing a Server

Use `describe` to display the contents of the server's configuration file. The meaning of each entry in the configuration file is described in Appendix B.

An example of output to `describe`:

```
Darwin SCU > describe DarthSmp-1
Architecture            : sparc
Server name             : DarthSMP-1
Server description      : One-node 37 on darth-smp
```

```
Server host name        : darth-smp.think.com
Listen port             : 1123
Executable              : /opt/TMCdarwin//bin/darwinhost-sparcsmp
Auto restart on reboot  : No
Number of nodes         : 1
Serial root             : $USERHOME/darwin
Distributed root        : Default
Shared project file     : No shared project file specified
Shared serial root      : No shared serial root specified
Shared distributed root : No shared distributed root specified
Environment entry       : LD_LIBRARY_PATH=/opt/TMCdarwin//lib/
Environment entry       : DARWINHOME=/opt/TMCdarwin/
Environment entry       : ORACLE_HOME=/opt/TMCdarwin/
Environment entry       : NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
Environment entry       : ORA_NLS33=/opt/TMCdarwin/ocommon/nls/admin/data
```

4.4.5 Verifying the Validity of a Configuration File

Use `verify` to confirm that the configuration file for the server is valid. The `verify` subcommand ensures that you can actually connect to specified ports and that any files mentioned in the configuration file actually exist. It also verifies that the configuration is appropriate for the specified architecture. You can use `verify` to ensure the validity of a configuration file that you have created with the subcommand `add`.

4.4.6 Starting a Server

Use `start` *`server-name`* to start the Darwin server with name *`server-name`*. You can only start configured servers; a server is not configured until you save the configuration file (using the `save` subcommand). `darwinconfig` verifies the server's configuration file and tries to start the server. If `darwinconfig` finds problems with the server's configuration file, it indicates the problem(s) and asks whether you want to start the server anyway. If `darwinconfig` cannot start the server, it tells you why.

Note: Be sure you are logged in as root when starting and stopping Darwin servers. If you are not root, you will be able to start and stop the server, but when another user tries to enter Darwin, the servers are automatically stopped.

The `darwinconfig` command is interactive; however, you can use the following one-line version of `darwinconfig` to start a configured Darwin server with name *`server-name`*:

```
# darwinconfig start server-name
```

If you use this one-line version of the command, `darwinconfig` always tries to start the server. You must be root when you execute this command.

4.4.7 Stopping a Server

Use `stop` *`server-name`* to stop the Darwin server with name *`server-name`*.. You can only stop servers that are running.

Note: Be sure you are logged in as root when starting and stopping Darwin servers. If you are not root, you will be able to start and stop the server, but when another user tries to enter Darwin, the servers are automatically stopped.

The `darwinconfig` command is interactive; however, you can use the following one-line version of darwinconfig to stop a configured Darwin server with name *`server-name`*:

```
# darwinconfig stop server-name
```

4.4.8 Saving a Configuration File

Use the `save` subcommand to write the information you have entered to Darwin's internal configuration files.

The subcommand add does not automatically save the configuration; you must use `save` to save the configuration file before you can start the server. You can also end the session using `exit`, which saves all changes before ending the session.

4.4.9 Ending a darwinconfig Session

There are two ways to end a `darwinconfig` session:

- `exit` saves changes, then ends the session.
- `quit` ends the session without saving changes.

4.4.10 Getting Help

Enter `help` or `?` at the `Darwin SCU >` prompt to get a list of the subcommands and a brief description of each.

4.4.11 Modifying and Adding Environment Variables

This section describes the following topics:

- Editing the default environment variables
- Adding environment variables that are not automatically created

4.4.11.1 Modifying Default Environment Variables

`darwinconfig` automatically adds the default value of all required environment variables to any server configuration. `darwinconfig` adds the following statements to all configurations:

```
env LD_LIBRARY_PATH=/opt/TMCdarwin/lib
env DARWINHOME=/opt/TMCdarwin/
env ORACLE_HOME=/opt/TMCdarwin/
env NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
env ORA_NLS33=/opt/TMCdarwin/ocommon/nls/admin/data
```

The default values for all of these variables, except the NLS_LANG variable, must be changed if Darwin server software is not installed in the default location `/opt/TMCdarwin`.

If Darwin server software is installed in the UNIX directory `/MyDarwinDir`, follow these steps to correct the values of the environment variables:

1. Create the server configuration using `darwinconfig` and save it.
2. Exit `darwinconfig` without starting the server.

3. Using the UNIX text editor of your choice, edit the configuration file so that `/opt/TMCdarwin` is replaced with `/MyDarwinDir`; when you are finished, the environment variables will read:

```
env LD_LIBRARY_PATH=/MyDarwinDir/lib
env DARWINHOME=/MyDarwinDir
env ORACLE_HOME=/MyDarwinDir
env NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
env ORA_NLS33=/MyDarwinDir/ocommon/nls/admin/data
```

4. Save your changes and exit the editor.
5. Use the `verify` command of `darwinconfig` to check the modified configuration.
6. Start the server.

4.4.11.2 Adding Environment Variables

Below is an example showing `darwinconfig add`'s prompts and user responses (shown in bold) for setting a Darwin environment variable required by a database package. In this example, Darwin server software is installed in `/MyInstallDirectory` instead of the default location. This example also shows the system's response when you press the RETURN key for help as prompted.

```
Would you like to add any environment variables to the server environment?
(y/n) : y

Environment variable("exit" to exit)(return for help) :
LD_LIBRARY_PATH<return>

The name of a environment variable to add to the Darwin server process
environment. This is necessary for most database connectivity setups. See your
database documentation for more information.

Environment variable ("exit" to exit)(return for help):
Value of "LD_LIBRARY_PATH" (return for help) : <return>

The value of a environment variable to add to the Darwin server
process environment. This is necessary for most database connectivity setups.

Value of "LD_LIBRARY_PATH" (return for help) :
/MyInstallDirectory/lib

Environment variable ("exit" to exit) (return for help) : exit
Done.
```

4.4.12 Modifying Default Configurations for Shared Servers

If you configure the server to allow shared projects, `darwinconfig` automatically adds the following two statements to the server configuration:

```
shared_project_file /opt/TMCdarwin/sharefile
shared_serial_root /opt/TMCdarwin/shared_projects
```

If you want to specify a different `sharefile` and/or a different `shared_serial_root` (that is, a different directory where shared projects are stored), save the configuration, exit `darwinconfig`, and edit the file using a UNIX text editor to change the pathnames. Before you start the server, run the `darwinconfig verify` command to check the modified configuration.

4.4.13 Modifying Configuration Files

You cannot modify configurations, except to edit environment variables for nonstandard installations and to change directories associated with shared servers, as described in Sections 4.4.11.1 and 4.4.12. For all other changes, you must remove the server (as described in Section 4.4.3) and configure it again, as described in Section 4.4.2. If the removed server has ever been used, you must give the server a new name when you add it. The reason it must have a new name is that names of servers that have been used are embedded in the `.darwinprojects` and `.darwinhostrc` files in users' home directories.

4.5 Troubleshooting Configurations

Here are solutions to common problems associated with server configurations:

1. Use `darwinconfig` to create configuration files; if you make changes using a text editor the changes may not work. In particular, configuration files are not parsed by any standard UNIX shell; among other things, dependent variables will not work.
2. If you change a configuration, make sure that you stop the server and restart it. Changes will not take effect until you restart the server.
3. If you change configuration parameters, you may have to reboot the server.
4. The Darwin executable must be running on the server.

5

Oracle Connectivity

Note: Darwin no longer uses ODBC to connect to Oracle databases; it uses Oracle Call Interface (OCI) instead.

If you will not be connecting to an Oracle database, you can skip this chapter.

Any Darwin server can connect to a properly configured Oracle database. This chapter describes setting up the environment for database connectivity. Follow these steps:

- Obtain, install, and connect to the database software (Section 5.3).
- Confirm / set permissions and file / directory accessibility (Section **5.3.2**).
- Create required files (Section 5.3.3).

Note: Existing `.odbc.ini` files that worked with releases of Darwin previous to Release 3.7 no longer work; they must be replaced with `.darwindb.ora` files.

- Check that the installation is correct and that the environment variables have been set correctly (Section 5.5).
- After you have installed the Darwin client software (Chapter 6), confirm that users can connect to the database software (Section 5.6).

For examples of configuration files for Darwin servers, see Appendix B.

After you configure database connectivity, you can connect to and disconnect from databases and create Darwin datasets from the result of an SQL script. Alternatively, you can use the **Database Import** Wizard to create a Darwin dataset from a

database table and the **Database Export** Wizard to store datasets and results tables in a database. You cannot perform any other operations, such as storing models in a database.

5.1 Upgrading Existing Server Configurations

Server configurations that worked with Darwin 3.6.1 or earlier versions of Darwin will not work with Darwin 3.7. Existing server configurations must be recreated. This chapter explains how to configure Oracle database connectivity. Appendix C summaries the changes required that are not related to database connectivity.

5.2 Oracle Call Interface

Darwin does not use ODBC drivers to connect to Oracle databases; at Release 3.7, Darwin uses Oracle Call Interface (OCI) to connect to Oracle databases.

OCI is an application programming interface that allows applications, such as Darwin, to use the native procedures or function calls of a third-generation language to access an Oracle database server and control all phases of SQL statement execution. OCI provides improved performance and scalability through the efficient use of system memory and network connectivity.

OCI for Darwin 3.7 requires the Oracle 8.0.6 OCI library; the required library is bundled with Darwin 3.7.

5.3 Database Requirements, Permissions, and Required Files

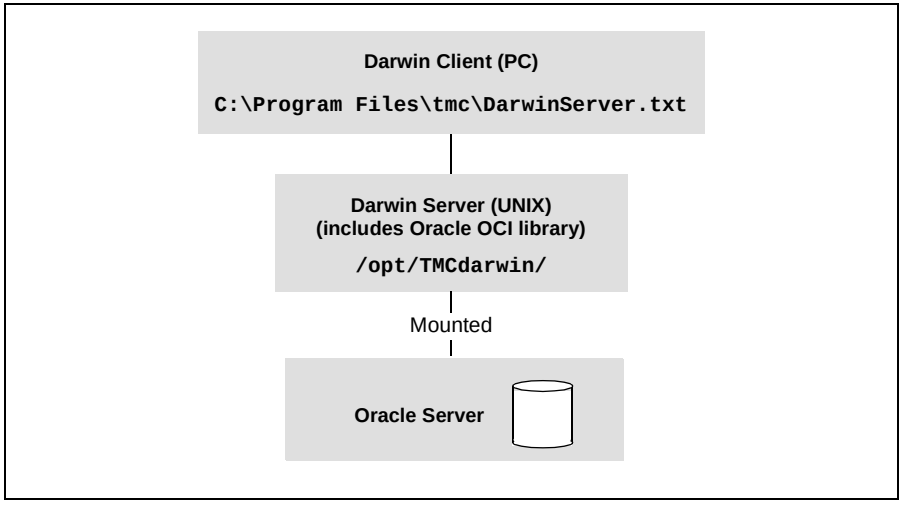
Darwin 3.7 supports connectivity to Oracle databases only. Darwin connects to:

- Oracle 8.1.6

If you do not already have Oracle database software on your system, you must contact Oracle Corporation, as described in Section **2.6.1**, to learn how to obtain and configure Oracle software.

Figure **5–1** illustrates database connectivity between Darwin and an Oracle database.

Figure 5–1 Connectivity between Darwin and Oracle.



In order to connect to an Oracle database from a Darwin server, you must

- Install Oracle database software and configure it to satisfy the requirements described in Section 5.3.1
- Set permissions set as described in Section 5.3.2
- Create the required files described in Section 5.3.3
- Configure a server as described in Chapter 4

5.3.1 Database Requirements

Any Oracle database that Darwin connects to must satisfy the following requirements:

- A listener must be set up, even if Darwin and the Oracle database are installed on the same system.
- Oracle must support single-byte 8-bit ASCII.
- The Oracle database must be a network accessible from the UNIX system where Darwin server software is installed.

5.3.2 Permissions for Oracle Databases

The following permissions are required to connect to Oracle databases from Darwin:

- Users must be recognized by the database server and must have database connect permission.
- Users must have read permission on the database tables.

5.3.3 Required Files

The following file is required:

- `.darwindb.ora`

If the Oracle database and the Darwin server are on different systems, the following file must be edited as described in Section 5.3.3.2:

- `/opt/TMCdarwin/network/tnsnames.ora`

5.3.3.1 .darwindb.ora File

Note: Existing `.odbc.ini` files are ignored.

Each user's home directory on UNIX must contain an appropriate `.darwindb.ora` file to identify the data sources.

Here is a sample `.darwindb.ora` file for Oracle connectivity. `ServerName` is the SERVICE_NAME in the file `/opt/TMCdarwin/network/admin/tnsnames.ora`, described in the next section.

```
[Oracle8iboth]
Description=Oracle8i - DB on dev-console
ServerName=myDB
```

5.3.3.2 tnsnames.ora File

Note: You must configure Listener for the Oracle 8.1.6 database that you connect to, even if it is installed on the same machine as Darwin server.

You must edit the file `/opt/TMCdarwin/network/admin/tnsnames.ora` (or `/InstallDir/network/admin/tnsnames.ora`, if Darwin server software is not installed in the default location) to specify the service name for the data source to which you will connect. For example, if `myDB` is accessible from port 1521 on `myHost`, /opt/TMCdarwin/`network/admin/tnsnames.ora` should contain:

```
myDB=
  (DESCRIPTION=
     (ADDRESS=
       (PROTOCOL=TCP)
       (HOST=myHost)
       (PORT=1521)
     )
     (CONNECT_DATA=
        (SERVICE_NAME=myDB
     )
  )
```

Note: Do not use tabs in `tnsnames.ora`.

5.4 National Language Characters in Tables

Oracle databases provide National Language Support (NLS) that permits users to store, process, and retrieve data in many languages. NLS ensures that database utilities, messages, sort order, date, time, currency, numbers, and calendar conventions automatically adapt to a specified language and locale.

If Darwin and an Oracle database specify different NLS settings, Oracle 8 converts strings automatically. During this conversion, information may be lost.

The locale-specific NLS data is stored in the directory specified in the ORA_NLS33 parameter. Darwin supports the setting `AMERICAN_AMERICA.WE8ISO8859P1`.

A national language supports territories (used to specify data, currency, and numeric formats) and character sets.

Darwin 3.7 supports the American language, the America territory, and the `WE8ISO8859P1` character set. A Darwin environment variable is automatically added to specify this NLS setting in all Darwin server configurations. Darwin does not support any other NLS setting.

You can import Oracle tables using the `WE8ISO8859P1` character set into Darwin, perform actions on the resulting dataset (transformations, model building, etc.), and then export resulting datasets back to Oracle tables.

5.5 Installation Check

Before you connect to an Oracle database using the Darwin client, check that your installation of Oracle and the OCI drivers is correct. To do so, log in to the machine where Darwin server software is installed and connect to the Oracle database using the Oracle utility `sqlplus`.

If you can complete this step successfully, you will be able to connect to Oracle databases from the Darwin client.

5.6 Using Database Connectivity

After you have installed the Darwin client software (Chapter 6), you will be able to use the Darwin client to connect to a database. This section describes the different ways a user can connect to a database from the Darwin client.

5.6.1 Importing Tables Directly

The following commands on Darwin's **Project** menu permit users to connect to a database server and to terminate an existing connection to a database server:

- **Database Connect**
- **Database Disconnect**

The **Database** tab of the **Dataset** menu's **Create** command lets users create a Darwin dataset from a connected database using an SQL script or query. For information about these operations, see *Using Darwin* and *Darwin Reference* or the Darwin online help. See Section 5.6.2 for another way to import data from a database.

5.6.1.1 SQL Scripts

Darwin allows you to run an SQL script after you have connected to a database. For more information and examples, see *Using Darwin* and the *Darwin Reference.*

5.6.1.2 SQL Queries

Darwin allows you to run an SQL query after you have connected to a database. For more information and examples, see *Using Darwin* and the *Darwin Reference*.

> **Note:** Do not terminate the query with a semicolon (;); if you terminate the query with a semicolon, the import operation will fail with an Oracle message about an invalid character. Also do not enter RETURN characters in the query.

5.6.1.3 SQL Limitations

When you use a script or query to create a new Darwin dataset, do not use the SQL UNION, INTERSECT, and EXCEPT operators or the COUNT column function; using these items results in an error.

5.6.1.4 SQL Formatting Limitation

All SQL commands in either scripts or queries must be on one line (that is, they must not contain RETURN characters). Commands that are on more than one line will either result in an message about invalid SQL or will not execute correctly. For example, the following statement in a script will execute correctly:

```
SELECT * FROM MYTABLE WHERE FIELD1 <> 5;
```

The following statement will not be executed correctly by Darwin; the WHERE clause will not be executed:

```
SELECT * FROM MYTABLE
         WHERE FIELD1 <> 5;
```

5.6.2 Database Import Wizard

The **Database Import** Wizard simplifies the process of converting a database table to a Darwin dataset. The Wizard performs the following functions:

- connects you to the database
- permits you to log in to the database
- displays available database tables
- converts the selected table to a Darwin dataset
- closes the connection to the database

To use the Wizard to import data, there must be a table that contains the data that you wish to import, and you must provide the name of that table.

The Wizard manages the connection to and the disconnection from the database automatically.

For more information about this Wizard, see *Darwin New Features.*

5.6.3 Database Export Wizard

The **Database Export** Wizard permits you to save a Darwin dataset or results table as a database table. You can write the dataset or result table to a new table in the database, or you can update or overwrite an existing database table. The Wizard

- connects you to the database
- lets you select a Darwin dataset
- exports the dataset to a database table
- closes the connection to the database

For more information about this Wizard, see *Darwin New Features.*

6

Installing the Darwin Client

After you've installed the Darwin server on Solaris and configured and started one or more Darwin servers, you are ready to install Darwin client software on PCs running Windows NT, Windows 95, or Windows 98.

Follow these steps to install Darwin client software:

1. Verify that all software requirements are satisfied (Section 6.1).
2. Gather information about the Darwin servers that you will connect to (Section 6.2).
3. Create a Darwin Windows Client customization disk or file (optional) (Section 6.2.1).
4. Install the client software (Section 6.3).

6.1 Darwin Client Software Requirements

Darwin client requires the correct regional settings, Internet Explorer, and Excel for complete operation.

6.1.1 Regional Settings

The regional settings on any PC running the Darwin client must be English (United States). If you specify any other language, the Darwin client will not operate properly.

To view regional settings, click **Start > Settings > Control Panel**; then click **Regional Settings** (globe icon). Go to the **Regional Settings** tab. The value must be `English (United States)`.

6.1.2 Microsoft Internet Explorer

Microsoft Internet Explorer 4.0 is required for the display of online help. If Internet Explorer is not installed on a PC where Darwin client is installed, no online help will be available. (**Help** buttons will not work; the **F1** key will not display help; the **Help** tab links will not work; the **Help** button on the tool bar will not work; all items in the **Help** menu, except for **Help > About Darwin**, will not work.)

Except for the lack of online help, Darwin installed on a PC without Internet Explorer will work properly.

See Section 2.3.3 for information about obtaining Internet Explorer 4.x.

6.1.3 Microsoft Excel

Microsoft Excel is used by the Darwin client to create plots and graphs; one of the following is required:

- Microsoft Excel 95 for Windows 95, Version 7.0
- Microsoft Excel 97

Note that Excel must be installed on the C Drive of your PC.

If Excel 95 or Excel 97 is not installed on your PC, you will not be able to create any plots or graphs using Darwin.

6.2 Collect Server Information

Before you install Darwin client software, you must have the following information for each server that the client will connect to:

- the name of the server
- the name of the host machine on which the server resides (the IP address for the E10000 partition, see the note below for details)

> **Note:** If a Darwin server is running on a partition of an E10000, the partition must have a static IP address; use this static IP address as the host name when you specify server information.

- the port number for the server
- a description of the server (optional)

During the installation process, you will be prompted for this information in a screen titled “Create the Configuration File”.

If you are connecting to several servers or are installing several clients, you may want to create a Darwin Windows Client customization disk or file (Section 6.2.1).

6.2.1 Create a Customization Disk or File (Optional)

If you plan to install Darwin client on several PCs, you may want to create a customization disk or file.

Follow these steps to create a Darwin Windows Client customization disk or file:

1. Using **Notepad** or a similar editor (not **WordPad**), create a file named `DarwinServers.txt`, containing one-line descriptions of the Darwin servers that you plan to connect to. For each server, create a line in the following format:

   `server` *`name host port description`*

   where *`name`* is the name of the server, *`host`* is the system on which the server resides, *`port`* is the port number for the server, and *`description`* is the optional description of the server, enclosed in quotation marks.

   Separate each item from the next with a space (not a TAB).

   For example, if you plan to connect to the server `eval-1` on the machine `test.mysite.com` at `port 606` with description `"Test System 1"`, include the following line in the `DarwinServers.txt` file:

   ```
   server eval-1 test.mysite.com 606 "Test System 1"
   ```

   This line is the entry for the server from the `.darwinservers` file; you can create the customization file or disk by copying the server entries in the `.darwinservers` file.

2. To create a Darwin Windows Client customization disk, copy `DarwinServers.txt` to a diskette. To create a customization file, copy `DarwinServers.txt` to a folder that can be seen by all the personal computers that will run Darwin client software. The `DarwinServers.txt` file also belongs in each user’s `C:\Program Files\Tmc\Darwin Windows Client` folder.

You can use the same Darwin Windows client customization disk or file for all supported PC platforms.

6.3 Install Client Software

The client software is installed using an InstallShield wizard. Follow these steps to install the client software:

1. Exit all programs that are running on the PC where you are installing a Darwin client.
2. Insert the Darwin Client CD in the CD drive of the PC where you are installing the Darwin client.
3. If you are using a Darwin Windows Client customization diskette, insert it into the diskette drive of the same PC.
4. The **Darwin Windows Client Setup Program** will start up automatically. The installation wizard will guide you through the installation process.
5. If the setup program does not start automatically, navigate to the CD ROM drive and double-click `Setup.exe`.
6. To complete the process, the PC must be restarted. You can either let the wizard restart the PC or you can restart the PC at your convenience. If you are using a customization disk, remove it before you restart the PC.

6.3.1 DARWINHOME Windows Environment Variable

During installation of the Darwin client, Darwin adds a Windows environment variable named `DARWINHOME`, and sets it to the directory in which you installed the Darwin client, which is by default `C:\Program Files\Tmc\Darwin Windows Client`.

This variable appears on the **Control Panel > System > Environment** tab under **User Variables**. It is possible to modify its setting there, but *do not*. If the setting is modified, Darwin is not likely to work at all.

6.4 First-Time Installation

Before you can use Darwin for the first time, you must create a least one project and copy at least one text file or dataset in that project. Until you create a project and copy a text file or dataset into it, everything in the Darwin graphical user interface is greyed out and no commands are available. There are two ways to do this:

- Copy a dataset or text file from the datasets for practice as follows:
  1. Set up the datasets for practice, as described in *Darwin New Features*.

2. Create a project as described in Section 8.2.
3. Drag a dataset from the datasets for practice into your project.
4. Commands will now be available.

- Create a project and copy a text file into it as follows:
  1. After both Darwin client and server software are installed, start up the Darwin client.
  2. Create a project (described in Section 8.2).
  3. Exit Darwin.
  4. On UNIX, go to the `darwin` directory, and, in the project directory for the project you just created, create a `.txt` file (or copy a `.txt` file into the directory), and, optionally, a `.des` file.
  5. When you attach to the Darwin client, commands will be available.

7

Starting and Stopping Darwin

After the Darwin server and client software are installed, and at least one Darwin server is configured and running, you and other users can start using Darwin. The system administrator will want to start up Darwin to confirm that Darwin is installed correctly and is ready for users.

7.1 Starting Darwin

Starting Darwin requires the following steps, which must be performed in the order indicated:

- First, the system administrator starts all necessary Darwin servers on UNIX (Section 7.1.1).
- Next, ordinary users start the client software on Windows NT or Windows 95/98 (Section 7.1.2).

7.1.1 Starting Darwin Servers

A system administrator uses the `darwinconfig start` *`server-name`* command described in Section 4.4.6 to start each server that a user will connect to.

Note: Be sure you are logged in as root when starting and stopping Darwin servers. If you are not root, you will be able to start and stop the server, but when another user tries to enter Darwin, the servers are automatically stopped.

7.1.2 Starting Darwin Clients

To start Darwin,

- Log in as an ordinary user on Windows NT, Windows 95, or Windows 98.
- Click the **Start** button; on the **Start** menu, click **Programs**; on the **Programs** menu, click **Darwin Windows Client** (identified by a protohominid icon); if there is a Darwin icon on your desktop, you can double-click it to start the client.

The Darwin splash screen appears. After the splash screen disappears, the Darwin login window appears (titled **Darwin Login**).

Now log in to a Darwin server. At the top of the login screen are two boxes for you to type in:

- **User Name:** Enter the user name for your account on the UNIX server (where Darwin server software is installed).
- **Password:** Enter the password for your account on the server. This password may or may not be the same password that you use to log on to your PC.
- **Darwin Server:** This text box displays a list of Darwin servers for you to choose from. (The list comes from the `DarwinServers.txt` file that you created when the client software was installed.) Click the name of the Darwin server you want to connect to. To the right of the server name is a brief description of it.
- Click **OK** or press ENTER.

Darwin then attempts to connect you to the server you selected, and the main Darwin window is displayed.

If the attempt to connect fails, try connecting to different server or contact your system administrator. The administrator can check to see whether servers are running and start them if necessary.

7.2 Stopping Darwin

Stop Darwin as follows:

- First, ordinary users exit (stop) all clients on Windows NT, Windows 95, or Windows 98 (Section 7.2.1).
- Then, the administrator stops all servers on UNIX (Section 7.2.2).

7.2.1 Stopping Darwin Clients

The best way to exit Darwin is to click **Project > Exit** (on the **Project** menu, click **Exit**). When you exit by this route, Darwin gives you a last opportunity to save any files or objects that you created in this session but did not save.

7.2.2 Stopping Darwin Servers

The system administrator stops each server using the `darwinconfig stop` *name* command described in Section 4.4.7.

Note: Be sure you are logged in as root when starting and stopping Darwin servers. If you are not root, you will be able to start and stop the server, but when another user tries to enter Darwin, the servers are automatically stopped.

8

Administering Darwin Projects

This chapter provides general information about Darwin projects; these are basic things system administrators should know to support Darwin users.

> **Note:** If you install Darwin on a system for the first time, you must take certain steps to make the GUI usable. See Section 6.4 for details.

8.1 Darwin Projects

All work in Darwin is done within the context of a project, which is a UNIX directory that resides on the server. Ordinary users create and delete projects.

At the beginning of your first Darwin session, the only commands available are those that create new projects (**New Project**), select existing projects (**Open Project)**, or end the session (**Exit**).

The first time that you run Darwin, you must create a project. After you have created at least one project, Darwin automatically selects the project you last used whenever you start a new Darwin session. You can work in that project, select another project, or create a new project.

For more information about projects, see Darwin's online help, *Using Darwin*, and *Darwin Reference*.

8.2 Creating Projects

Ordinary users create projects from the Darwin client, using the **New Project** command of the **Project** menu.

Follow these steps to create a project:

- On the **Project** menu, click **New Project**.

A dialog appears, and prompts you for the following information:

- In the **Project** box:
  - **Name:** Specify a name for the project.
  - **Description:** (Optional.) Record any information you may want about the project.
  - **Shared:** If you want this project to be shared with other users, click the **Shared** box. This option is available only if the Darwin server is configured to permit shared projects
  - The **In Server** box contains two read-only boxes: **Name**, the name of the current server, and **Description**, the description of the current server.
  - **Distributed:** If you want the project to be distributed, click the **Distributed** box. This option is available only if the Darwin server was configured as a distributed server.
- If you wish, click **Advanced**, which takes you to **Advanced Options**, **Project** tab, where you can record information about the project (see Section 8.2.1, below).
- Click **OK** or press ENTER.

Darwin then creates the project, using the pathname or pathnames specified in the server configuration file.

8.2.1 Recording Project Information

If you click **Advanced Options** on the **New Project** dialog window, you can specify the following optional information about the current project (the project you are creating):

- **Leader:** The name of the project leader.
- **Phone** and **Ext:** The telephone number of the project leader.
- **Business Problem:** The problem that the project addresses.
- **Objectives:** The objectives of the project.

Supplying any of this information is optional.

When you are finished, click **OK** or press ENTER to save the information.

8.3 Deleting Projects

Ordinary users can delete their own projects by right-clicking the project's name in the **Workspace** listing; then, on the context menu that appears, click **Delete**.

You must delete all files associated with a project before you delete the project. If you try to delete a project with files left in it, you get one of these error messages:

```
Serial directory is not empty
```

or

```
Distributed project directory is not empty
```

8.4 Administrator Decisions about Projects

All project directories reside on the UNIX server.

The system administrator defines the location of Darwin project directories within the server configuration files (described in Chapter 4). The system administrator can also set up projects for users. Usually, however, users define projects for themselves.

Darwin supports two types of projects: user projects and shared projects:

- *User projects* are designed for use by a single user. Their UNIX umasks are set to 022.

  If the administrator uses the USERHOME keyword in the serial path defined in a server's configuration file, then project directories are subdirectories of the home directory of the user who creates them. If the administrator does not use the USERHOME keyword during server configuration, project directories are grouped as subdirectories of a single specified directory.

- *Shared projects* are designed for use by groups of users. Their UNIX umasks are set to 0, allowing all users to create, open, use, and delete files (if the base directories have write permission set for all users). As part of configuring a server, the system administrator specifies whether users can create shared projects. The USERHOME and USERNAME keywords are not allowed in shared pathnames.

Administrators can choose whether to allow shared projects on any given Darwin server. If they define shared serial root, shared project file, and shared distributed root (for distributed servers only) in a server's configuration file, then shared projects are allowed. If the shared files are not defined, then shared projects cannot be created on that server. See the discussion of `shared_serial_root` in Section B.2 for more information.

8.5 The .darwinprojects File

When Darwin creates a nonshared project for a user, it puts the project name and information in a `.darwinprojects` file, which is stored in the user's home directory on UNIX. The file contains one entry for each project the user creates.

Darwin creates and maintains the `.darwinprojects` file. Under ordinary circumstances, neither the user nor the administrator needs to edit the file.

When the user gives the **Project** menu's **Open Project** command, Darwin displays the project names from the user's `.darwinprojects` file. If the server has a shared projects file, those project names are also displayed.

The user selects a project from the list. Darwin then uses the `.darwinprojects` file to locate the project directories for this project, when running on this server. (If the project has not been run on this particular server before, Darwin goes to the server configuration file for default directory information.)

When you restart Darwin, it automatically selects the project you used last.

8.5.1 A Sample .darwinprojects File

As an example, the project `ad_campaign_3` might have the following entry in a `.darwinprojects` file:

```
project_name ad_campaign_3
directory /users2/csmith/darwin/ad_campaign_3
server eval-1 T /users2/csmith/darwin/ad_campaign_3
server RingD F /export/darwin/csmith/ad_campaign_3
end_project
```

The information for each project is as follows:

- the keyword `project_name`, followed by the project's name
- the keyword `directory`, followed by the pathname of the serial project directory

- the keyword `server`, followed by three items:
  - the name of a server on which the project has run
  - the keyword T or F, indicating that serial and distributed directories are in the same directory (T) or are in different directories (F)
  - the pathname for the distributed directory

  If the project has been run on more than one server, the file contains one line for each server that has been used.
- the phrase `end_project`

8.5.2 Modifying the .darwinprojects File

There are a few circumstances under which the user or administrator might want to alter the `.darwinprojects` file. For example,

- Whenever any directory containing Darwin projects is moved, the user or administrator should check the pathnames in the user's `.darwinprojects` file and correct any pathname affected by the move.
- The system administrator can change the location of the serial or distributed directories for a server by manually editing that server's configuration file. These changes do not propagate into the `.darwinprojects` files. This can be advantageous: For instance, it allows existing projects to remain on one set of disks, while new projects are placed by the server on new disks. On the other hand, if you want a user's projects to migrate to a new disk, then either the user or the administrator must edit the `.darwinprojects` file for that user.
- If two users want to share access to a user project, they can do so by copying the project's entry from the `.darwinprojects` file into the second user's `.darwinprojects` file. Permissions on the project directory may have to be modified; other modifications may be necessary.

A

Installing the SAS Conversion Utilities

Darwin supports importing and exporting datasets in proprietary formats using a combination of a third-party product (DBMS/COPY) and two Perl scripts distributed with Darwin (`sas2darwin` and `darwin2sas`). The Perl scripts automate the use of DBMS/COPY to convert files to and from the format supported by the SAS Institute, Inc.

DBMS/COPY, Perl, and the Perl scripts all run on UNIX; you cannot invoke the SAS conversion utilities from the Darwin client.

This appendix describes how to install the conversion utilities on UNIX. For information about the hardware and software requirements of the SAS conversion utilities, see Section 2.5 of this manual. For information about using the utilities, see *Using Darwin*.

A.1 Installing the Software

If you do not plan to convert to and from SAS files, you should not install the SAS conversion utilities.

Install DBMS/COPY and Perl as directed by the suppliers of the software.

The Perl scripts `sas2darwin` and `darwin2sas` are part of Darwin 3.6.1 and are installed automatically in the UNIX directory `/opt/TMCdarwin/bin` (`DARWINHOME/bin`) when Darwin is installed.

`sas2darwin` and `darwin2sas` assume that Perl is installed on UNIX in `/usr/local/bin/perl`. If Perl is not installed in this location on your system, you must edit the first line of the script `sas2darwin`, changing the line

```
#!/usr/local/bin/perl
```

to specify the location of Perl on your system. (`darwin2sas` is a link to `sas2darwin`, so you don't have to edit both scripts.)

B

Server Configuration Files

This appendix includes sample Darwin server configuration files and explains what the entries mean.

You create configuration files using the UNIX command `darwinconfig`, as described in Chapter 4.

Note: All file locations assume a standard installation in which the software is installed in `/opt/TMCdarwin`, the default location. If the software is installed somewhere else, replace `/opt/TMCdarwin` with the appropriate pathname.

B.1 Darwin Configuration Files

`darwinconfig` creates and maintains the following UNIX files:

- the file `.darwinservers`, which lists all configured Darwin servers at your site. `.darwinservers` is created in `/opt/TMCdarwin/etc`. Section 4.2.1 describes the `.darwinservers` file.
- a configuration file for each server on the system. Each server's configuration file has the same name as its server. These configuration files are found on UNIX in `/opt/TMCdarwin/etc`. Section B.1.1 shows a sample configuration file for each supported architecture running Solaris 2.7 and a sample configuration file for a server supporting database connectivity.

For more information about configuring and maintaining Darwin servers, see Chapter 4.

B.1.1 Server Configuration Files

The server's configuration file contains all the information necessary to run the server and set up projects. Configuration files are slightly different for each different

platform. The following sections contain sample configuration files for the Solaris platforms that Darwin supports at Release 3.7. Configuration files are created using the UNIX command `darwinconfig`, as described in Chapter 4. In the default installation, configuration files reside in the directory `/opt/TMCdarwin/etc`.

Changes in Configuration Files

See Appendix C for a summary of the changes that are required to upgrade servers that were created using versions of Darwin prior to 3.7.

The primary change for Darwin 3.7 is using OCI for database connectivity; the required library is bundled with Darwin. Any Darwin environment variables specifying these libraries must be corrected.

Sample Configuration File for a Single-Node Server

Here is the configuration file for a server named `eval-1` that runs on an Ultra-SPARC workstation. This server supports database connectivity if there is a properly configured Oracle database available; if no such database is available, the server does not support dababase connectivity. (For more information about database connectivity, see Chapter 5.) For a detailed explanation of the various terms, see Section B.2.

```
DARWIN(tm) DAEMON CONFIG
architecture sparc
name OneNode-NODB
description No database with 1 node
server MySol
port 1123
executable /opt/TMCdarwin/bin/darwinhost-sparcsmp
auto_restart true
nnodes 1
serial_root $USERHOME/darwin
distributed_root Default
env ORACLE_HOME=/opt/TMCdarwin/
env LD_LIBRARY_PATH=/opt/TMCdarwin/lib/
env DARWINHOME=/opt/TMCdarwin/
env NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
env ORA_NLS33=/opt/TMCdarwin/ocommon/nls/admin/data
```

All entries are required for the server to work properly. Note that the definition of `ORACLE_HOME` is required even when the server does not connect to a database.

Sample Configuration File for a Two-Node Server

Here is a configuration file for a two-node Darwin server on an Ultra Enterprise (SMP) system that supports connection to a properly configured Oracle 8i database. Chapter 5 contains detailed information about database configuration.

```
DARWIN(tm) DAEMON CONFIG
architecture sparc
name OneNodeDB
description Oracle 8i One node
server MySmp
port 1119
executable /opt/TMCdarwin//bin/darwinhost-sparcsmp
auto_restart true
nnodes 2
serial_root $USERHOME/darwin
distributed_root Default
env LD_LIBRARY_PATH=/opt/TMCdarwin/lib
env DARWINHOME=/opt/TMCdarwin/
env ORACLE_HOME=/opt/TMCdarwin/
env NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
env ORA_NLS33=/opt/TMCdarwin/ocommon/nls/admin/data
```

All entries are required for the server to work properly.

Sample Configuration File For Shared Projects

The following configuration is for a two-node server that supports database connectivity in the presence of an appropriately configured Oracle database and shared projects. Shared projects are stored in `/opt/TMCdarwin/shared_projects/`.

```
DARWIN(tm) DAEMON CONFIG
architecture sparc
name 2nodes-shared
description 2 nodes shared with OCI connect
server mySMP
port 1065
executable /opt/TMCdarwin//bin/darwinhost-sparcsmp
auto_restart true
nnodes 2
serial_root $USERHOME/darwin
distributed_root Default
shared_project_file /opt/TMCdarwin//sharefile
shared_serial_root /opt/TMCdarwin//shared_projects
shared_distributed_root Default
env LD_LIBRARY_PATH=/opt/TMCdarwin//lib/
env DARWINHOME=/opt/TMCdarwin/
```

```
env ORACLE_HOME=/opt/TMCdarwin/
env NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
env ORA_NLS33=/opt/TMCdarwin/ocommon/nls/admin/data
```

All entries are required for the server to work properly.

B.2 What Configuration Files Mean

This section lists all the entries that can appear in a configuration file for a Darwin server and explains what they mean.

`architecture` is one of the architectures supported by Darwin. Configurations created using any version of Darwin other than 3.7 may not specify the correct architecture. `darwinconfig` prints a list of the currently supported architectures, which are

- **`sparc`**: Sun Microsystems UltraSPARC workstations or Ultra Enterprise Servers running Solaris 2.7 (single-node or multinode servers with or without database connectivity)
- **`hpux`**: Hewlett-Packard 9000 Series 700 or Series 800 single-node or multinode systems running HP-UX 11.0 with or without database connectivity

> **Note:** If you intend to connect to an Oracle database, you must have Oracle software installed and correctly configured. There must also be an appropriate `.darwindb.ora` file in your home directory on UNIX. (See Chapter 5 for details.)

You must specify an architecture. (You specify a multinode or single-node server using the **`nnodes`** entry.)

`name` is the name of the server (that is, configuration) to be displayed in the list of choices presented to the user upon entering Darwin. The name is also used in the `.darwinservers` file, which `darwinconfig` creates and updates automatically. `name` is also the name of the configuration file that describes the server. This item is required for all types of servers. Two different servers cannot have the same name. If a server has been used, you cannot create a new server with the same name, even if you remove the old server, since the name is embedded in `.darwinprojects` and `.darwinhostrc` in users' home directories.

`description` is the text describing the server that is displayed to users on the **Darwin Login** screen. This item is required for all types of servers.

server is the hostname of the workstation or server where the Darwin server will run. This address plus the port address are used to connect users to the Darwin server. You can also specify the IP address of the workstation or server for this item. This item is required for all types of servers.

port is the TCP port number where the daemon listens for connect requests. For security reasons, the port number should be less than 1024 if the daemon is running as root and greater than 1024 if the daemon is running under a specific username. If several users will share the server, the daemon must run as root. The port number specified here must not be used by any other installed applications as described in `/etc/services` or the services map if using NIS or NIS+. This item is required for all types of servers.

> **Note:** The port number must be greater than 1024.

executable is the path to the Darwin host executable that the daemon will invoke when a user selects this server at the start of a Darwin session. The executable is added to the configuration file automatically. Configurations created using any version of Darwin other than 3.7 may not specify the correct executable.The executable must be one of the following:

- `/opt/TMCdarwin/bin/darwinhost-hpuxpar` (for HP-UX)
- `/opt/TMCdarwin/bin/darwinhost-sparcsmp` (for Solaris)

This item is required for all types of servers.

auto_restart specifies that this Darwin server is automatically restarted when the system reboots. All daemons started at reboot will run as root. You must be running `darwinconfig` while logged on as root to specify `auto_restart`. You can use the `list` subcommand of `darwinconfig` to see whether `auto_restart` is specified for a particular server. (Be sure to save the configuration before you list it or reboot the server host.) This item is required for all types of servers.

nnodes is the number of nodes on the server that the Darwin server will use. This number must be greater than or equal to 1 and less than or equal to the total number of CPUs that Darwin will use. This item is required for all servers.

The following two *keywords* can be used when you specify `distributed_root` and `serial_root`:

- **USERHOME** — Used to specify that project directories are created in a subdirectory of the user's home directory on UNIX. For example, if serial root is specified as `$USERHOME/darwin` and if the user with home directory

/users5/csmith has a subdirectory /users5/csmith/darwin, and created the project ad_campaign_3, the serial directory for that project would be /users5/csmith/darwin/ad_campaign_3.

- **USERNAME** — Used to specify that a user's projects are stored in UNIX directories that contain the name under which a user is logged in. Darwin creates a subdirectory for each user and places the user's projects in that subdirectory. For example, if the distributed root is /export/darwin/$USERNAME, and the user jjones creates the project proj, the distributed directory for that project would be /export/darwin/jjones/proj.

Note that you *cannot* use these keywords to specify shared roots.

serial_root is the base directory for users' nonshared serial project directories. (These are directories in the UNIX file system, usually under the users' home directory.) This item is automatically set to the directory that you specify in response to the "Root directory for projects:" prompt in darwinconfig add.This item is required for all types of servers.

> **Note:** When you configure a Darwin server, make sure that the partition where the serial_root directory resides is large enough to hold the datasets that you intend to use.

distributed_root is the base directory for all distributed directories created for projects run by this server. On any server running on a Solaris or HP-UX system, the entry for distributed_root is automatically set to Default, signifying that the distributed root is the same as the serial root.

shared_serial_root, **shared_distributed_root**, and **shared_project_file** are necessary only if you want to allow shared projects on a server. (Shared projects are visible to all users of Darwin on the server. Anyone using a shared project has full access to all project files.) You cannot use the keywords USERHOME and USERNAME when you specify these items. darwinconfig automatically creates these entries and sets them to default values. For information about changing the defaults, see Section 4.4.12.

- **shared_serial_root** is the base directory for shared projects' serial directories. The directory must be writable by all Darwin users. Darwin places the serial part of all shared projects in shared_serial_root. The default value is

  ```
  shared_serial_root /opt/TMCdarwin/shared_projects
  ```

- **shared_distributed_root** is the base directory for all distributed directories created for shared projects run by this server. For servers running on a Solaris system, the entry for `shared_distributed_root` is `Default`, signifying that the shared distributed root is the same as the shared serial root. This default value should not be changed.
- **shared_project_file** is the pathname of a file that lists all the shared projects that are available to users of Darwin on this server. It contains information in the same format as the `.darwinprojects` file. This file must be writable by all users creating shared projects. The default value is

  ```
  shared_project_file /opt/TMCdarwin//sharefile
  ```

env entries are added to the environment of the executable. Default values for all required environment variables are automatically added to the server configuration.

> **Note:** If Darwin server software is *not* installed in `/opt/TMCdarwin` (the default location), you must correct the default values of the environment variables. See Section 4.4.11 for details, including a list of the default variables.

C

Upgrading Darwin

You can ignore this appendix unless you have Darwin 3.6.1 or earlier installed on your system.

To upgrade from Darwin 3.6.1 or earlier, you must

- Install the new server software.
- Upgrade existing Darwin server configurations.
- Install the new client software.
- If you are upgrading from a version of Darwin earlier than Darwin 3.5, you must recreate models that you wish to use with Darwin 3.7.

C.1 Upgrading Datasets and Models

Datasets created using Darwin 2.x do not work with Darwin 3.7. Before you deinstall Darwin, export them and then recreate them using Darwin 3.7.

Datasets created using Darwin 3.0.1 or later will work with Darwin 3.7. Models created using Darwin 3.0 or Darwin 3.0.1 will *not* work with Darwin 3.7. You must recreate the models using Darwin 3.7.

Models and datasets created using Darwin 3.5 or later versions of Darwin do not need any conversion.

C.2 License Files No Longer Required

If you have a license for Darwin 3.x, you do not have to delete the license. Darwin no longer requires license files; existing license files are ignored.

C.2.1 License File Environment Variable

Darwin no longer requires a license file. If the `TMC_LICENSE_FILE` environment variable is defined, it is ignored.

C.3 Upgrading Darwin Server Configurations

Any configuration files created using a release of Darwin earlier than 3.7 may specify any of the following:

- an unsupported architecture
- an executable that no longer exists
- the wrong libraries for database connectivity
- no `ORACLE_HOME` environment variable or the wrong value for `ORACLE_HOME`

You must create a new configuration for each existing Darwin server.

Note that you *cannot* modify existing configurations files; instead you must create new configurations. See Section 4.4.13 for details.

C.3.1 Darwin 3.7 Server Architectures

The list of supported Darwin server architectures has changed. The supported 3.7 server architectures are

- **`sparc`** (Solaris sever with or without database connectivity)
- **`hpux`** (HP-UX server with or without database connectivity)

> **Note:** The architectures **`sparc`** and **`hpux`** do not have the same meaning that they did in releases of Darwin previous to the 3.7 release.

Existing server configurations must be recreated to reflect the supported architectures.

The following architectures are *not* supported in Darwin 3.7:

- **`sparcsmp`**
- **`hpuxpar`**
- **`sparcdb`**
- **`hpuxdb`**

C.3.2 Darwin 3.7 Server Executable

Server configurations must contain the following executable line:

```
executable /opt/TMCdarwin/bin/darwinhost-sparcsmp
```

(assuming that Darwin is installed in `/opt/TMCdarwin`). If any other executable is specified the server will not start. This line is automatically generated during server configuration. If the server configuration was created using Darwin 3.6.1 or earlier releases, this line many not be correct.

C.3.3 Environment Variables

All server configurations must define the following environment variables:

- `ORACLE_HOME` (the default value is `/opt/TMCdarwin`)
- `LD_LIBRARY_PATH` (the default value is `/opt/TMCdarwin//lib/`)

The default definitions are automatically added to configurations created using Darwin 3.7. These environment variables are required for all servers, even if the server does not support Oracle connectivity. Configurations created with versions of Darwin other than 3.7 do not specify the correct values of these environment variables. See Section 4.4.11 for details.

C.4 Upgrading Configurations Connecting to Oracle Databases

Darwin 3.7 uses Oracle Call Interface (OCI) to connect to Oracle databases; Darwin 3.6.1 and earlier used ODBC. Existing server configurations supporting database connectivity will not work with Darwin 3.7; all such configurations must be recreated.

For detailed information about configuring an Oracle database so that Darwin servers can connect to it, see Chapter 5.

The following list summarizes the items in the server configuration that must be corrected:

- `ORACLE_HOME` must be set to the location of the Darwin libraries (`/opt/TMCdarwin`, if Darwin is installed in the default location).
- The environment variable `LD_LIBRARY_PATH` must be set to the location where Darwin libraries are installed.

Requirements for the database that Darwin 3.7 connects to have changed, too:

- Darwin connects to Oracle 8.1.5 or 8.1.6.
- A listener must be set up, even if Darwin and the Oracle database are installed on the same system.
- Oracle must support 8-bit ASCII.

Existing `.odbc.ini` files that worked with releases of Darwin previous to Release 3.7 no longer work. Each user's home directory on UNIX must contain, instead, an appropriate `.darwindb.ora` file to identify the data sources.

Index

Symbols

A

C

D

E

F

H

I

K

L

M

N

O

P

Q

R

S

T

U

V